WHAT IS CLAIMED IS:

1. A method for assigning input variables between a feeder logic block and a receiver logic block to implement a logic function, wherein the input width of the logic function exceeds the maximum input width for both the feeder logic block and the receiver logic block, the method comprising:

providing a cost function that assigns a first cost to the number of product terms cascaded from the feeder logic block to the receiver logic block and also assigns a second cost that increases as the number of input variables assigned to the receiver logic block approaches its maximum input width; and

testing the cost function using a plurality of input variable assignments to determine an optimal input variable assignment.

2. The method of claim 1, further comprising configuring the receiver and feeder logic blocks to implement the logic function using the optimal input variable assignment.

3. The method of claim 1, wherein the cost function provided by the providing act also assigns a third cost that increases as the number of input variables assigned to the feeder logic block approaches its maximum input width.

4. The method of claim 1, wherein the first cost is an integer multiple of the number of product terms cascaded from the feeder logic block to the receiver logic block.

5. The method of claim 4, wherein the integer multiple is five.

6. The method of claim 1, wherein the second cost is an exponential function of the difference between the number of input variables assigned to the receiver logic block and its maximum input width.

7. The method of claim 6, wherein the second cost equals $(k_r - (K-\beta))^{\gamma}$, k_r being the number of input variables assigned to the receiver logic block, the maximum input width is K, and β and γ are positive integers.

8. The method of claim 7, wherein β and γ equal three.

9. A method for partitioning a plurality of logic functions amongst segments of a programmable logic device, wherein each segment comprises a plurality of logic blocks arranged from a first logic block to a last logic block, the plurality of logic blocks being configured to support a cascade chain extending from the first logic block to the last logic block for increasing the input width of cascaded product terms, and wherein each logic block has a maximum input width, the method comprising:

identifying split functions within the logic functions whose input width exceeds the maximum input width for the logic blocks;

providing a cost function that assigns a first cost to the number of outputs provided to each segment by the remaining segments and that assigns a second cost to the number of split functions within each segment; and

testing the cost function with a plurality of partitions of the logic functions amongst the segments to identify an optimal partitioning.

10. The method of claim 9, wherein the first cost is proportional to the number of inputs provided to each segment by the remaining segments.

11. The method of claim 9, wherein the second cost is an exponential function of the number of split functions within each segment.

12. The method of claim 11, wherein the second cost equals $\lambda \sum_i S_i^2$, wherein S_i is the number of split functions in the ith segment and wherein λ is a positive integer.

13. The method of claim 12, wherein $\lambda = 100$.

14. A processor for configuring a programmable logic device to implement a desired logic function, wherein the programmable logic device includes a plurality of logic blocks, each logic block having a maximum input width, the processor comprising:

a software fitting engine operable to identify functions within the desired logic functions whose input width exceeds the maximum input width such that the identified functions must be split between a feeder logic block and a receiver logic block within the programmable logic device, the software fitting engine being further operable to provide a cost function that assigns a first cost to the number of product terms cascaded from the feeder logic block to the receiver logic block and

also assigns a second cost that increases as the number of input variables assigned to the receiver logic block approaches its maximum input width, the software fitting engine being operable to test the cost function using a plurality of input variable assignments to determine an optimal input variable assignment.